Distributed by:

JAMECO® ELECTRONICS

www.Jameco.com ✦ 1-800-831-4242

The content and copyrights of the attached material are the property of its owner.

Product Specification

Z 80-SIO
Z 80A-SIO

Jameco Part Number 35684

General Description

The Z80-SIO (Serial Input/Output) is a dual-channel multi-function peripheral component designed to satisfy a wide variety of serial data conimunications requirements in microcomputer Systems. Its basic function is a serial-to-parallel, parallel-to-serial converter/controller, Jul-within that role-it is configurable by Systems software so its "personality" can be optirnized for a given serial data cornrnunications application.

The Z80-SI0 is capabie of handling asynchronous formats. synchronous byte-oriented protocols such as IBM Bisync, and synchronous bit-oriented protocols such as HDLC and SDLC. This versatile device can also be used to support virtually any other serial protocol for applications other than data comrnunic'ations (cassette or floppy disk interfaces. for example).

The Z8O-SI0 can generate and check CRC Codes in any synchronous mode and can be programmed to check data integrity in various modes. The device also has facilities for modern controls in both channels. In applications where these controls arc not needed, the modern controls can be used for general-purpose I/O.

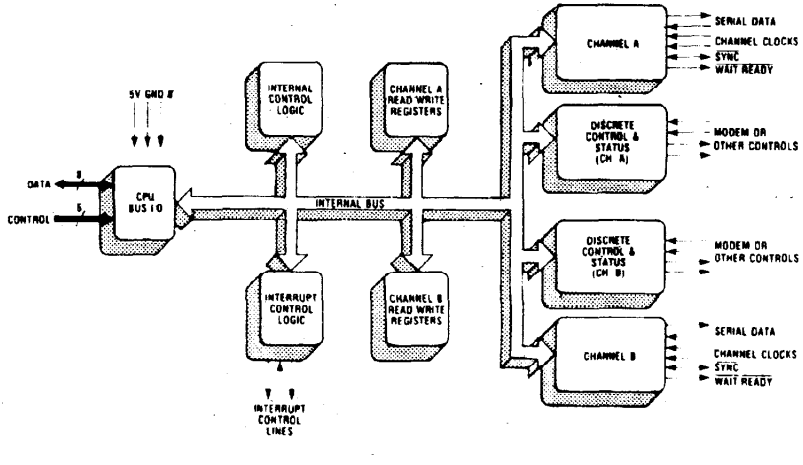
Structure

- N-channel silicon-gate depletion-load technology
- 40-pin DIP
- Single SV power supply
- Single-Phase 5 V clock
- All inputs and Outputs TTL compatible

Features

- Two independent full-duplex channels
- Data rates in **synchronous or isosynchronous rnodes:**
  - 0-50OK bits/second with 2.5 MHz System clock rate
  - O-800K bits/second with 4.0 MHz System clock rate
- Receiver data regisrers quadruply buffered; transrnitter doubly buffered.
- Asynchronous features:
  - 5, 6. 7 or 8 bits/character

Fig. 8 - SI0 BLOCK DIAGRAM



Z 80-SIO
Z 80A-SIO

Z-80 SIO Pin Description

- 1, 1½ or 2 srop bits
- Even. odd or no parity
- x 1, x 16, x32 and x64 clock modes
- Break generation and detecrion
- Parity. overrun and framing error detection

■ Binary synchronous features:
- Internal or external Character synchronization
- One or two sync characters in separate registers
- Automatic sync Character insertion/deletion
- CRC generation and checking

■ HDLC and SDLC features:
- . Abort sequence generation and detection
- Automatic zero insertion and deletion
- Autbmatic flag insertion between messages
- Address'field recognition
- . Support for one to eight bits/character
- Valid receive messages protected from overrun
- . CRC generation and checking

● Interrupt features:
- Daisy-chain interrupt logic provides automatic interrupt vectoring with no external logic
- Prograrnmable interrupt vector
- Status Affects Interrupt Vcctor mode for fast interrupt processing

■ CRC-16 or CRC-CCITT block frame check

■ Separate modem control inputs and Outputs for both channels

■ Modem status can be monitored

Pin Description

D_0-D_7. *System Data Bus* (bidirectional. 3-state). The system dara bus transfers dara and commands between the CPU and the Z80-SIO. D_0 is rhe least significant bit.

B/$\overline{A}$. *Channel A Or B Select* (input, High selects Channel B). This input defines which channel is accessed during a data transfer between the CPU and the Z80-SIO. Address bit Ao from the CPU is oftcn used for the selection function.

C/$\overline{D}$. *Control* Or *Data Select* (input, High selects Control). This input defines the type of information transfer pcrformed between the CPU and the Z80-SIO. A High at this input during a CPU write to the Z80-SIO causes the information on the data bus IO be interpreted as a command for the channel selected by B/$\overline{A}$. A Low at C/$\overline{D}$ means rhat the information on the data bus is data. Address bit A_1 is often used for this function.

$\overline{CE}$. *Chip Enable* (input. activc Low). A Low level at this input enables the Z80-SIO to accept command or data input from the CPU during a write cycle, or co transmit data IO the CPU during a read cycle.

ϕ. *System Clock* (input). The Z80-SIO uses the standard Z80 System Clock to synchronize internal Signals. This is a single-phase clock.

Mi. *Machine Cycle One* (input from ZSO-CPU, active Low). When $\overline{M1}$ is active and $\overline{RD}$ is also active, the Z80-CPU is fetching an instruction from memory; when $\overline{M1}$ is active while $\overline{IORQ}$ is active, the Z80-SIO accepts $\overline{M1}$

FIG. 9 – Z80-SIO/0 PIN CONFIGURATION

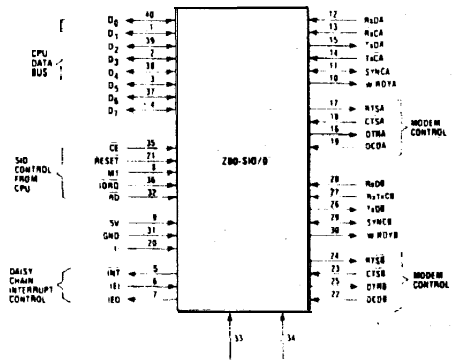
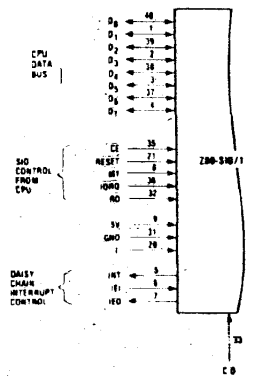


FIG.10 – Z80-SIO/1 PIN CONFIGURATION



Z-80 SIO Pin Description

and $\overline{IORQ}$ as an interrupt acknouledge if the Z80-SIO is the highest priority device that has interrupted the Z80-CPU.

IORQ. *Input/Output Request* (input from CPU, active Low). IORQ is used in conjunction with B/$\overline{A}$, C/$\overline{D}$, $\overline{CE}$ and $\overline{RD}$ to transfer commands and dara beiween the CPU and the Z80-SIO. When $\overline{CE}$, $\overline{RD}$ and $\overline{IORQ}$ are all active, the channel selected by B/$\overline{A}$ transfers data to the CPU (a read Operation). When $\overline{CE}$ and $\overline{IORQ}$ are active, but $\overline{RD}$ is inactive, the channel selecred by B/$\overline{A}$ is written to by the CPU wich either data or control information as specified bv C/$\overline{D}$. As mentioned oreviouslv. if $\overline{IORQ}$ and $\overline{M1}$ are active simultaneously. the CPU is acknowledging an interrupt and the Z80-SIO automatically places its interrupt vector on the CPU data bus if it is the highest priority devicc requesting an interrupt.

$\overline{RD}$. *Read Cycle Status*. (input from CPU, active Low). If $\overline{RD}$ is active. a memory or I/O read Operation is in progress. $\overline{RD}$ is used with B/$\overline{A}$, $\overline{CE}$ and $\overline{IORQ}$ Co transfer data from the Z80-SIO to the CPU.

$\overline{RESET}$. *Reset* (input. active Low). A Low RESET disables both receivers and transmitters, forces TxDA and TxDB marking. forces the modern controls High and disables all interrupts. The control registers must be rewritten afta the Z80-SIO is reset and before data is transmitted or received.

IEI. *Interrupt Enable* In (input, active High). This signal is used with IEO to form a priority daisy chain when there is more than one interrupt-driven device. A High on this line indicates that no other device of higher priority is being serviced by a CPU interrupt Serviceroutine.

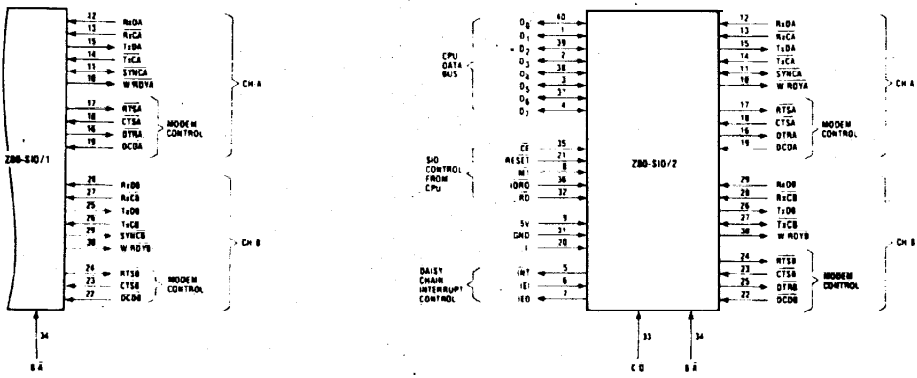
IEO. *Interrupt Enobk Out* (output, active High). IEO is High only if IEI is High and the CPU is not servicing an interrupt from this Z80-SIO. Thus, this signal blocks lower priority devices from interrupting while a higher priority device is being serviced by irs CPU interrupt service routine.

$\overline{INT}$. *Interrupt Request* (output, open drain, active Low). When the Z80-SIO is requesring an interrupt, it pulls $\overline{INT}$ Low.

$\overline{W/RDYA}$, $\overline{W/RDYB}$. *Wait/Ready A, Wait/Ready B* (Outputs, open drain when programmed for Wait function, driven High and Low when programmed for Ready function). These dual-purpose Outputs may be programmed as Ready lines for a DMA controller or as Wait lines that synchronize the CPU to the Z80-SIO data rate. The reset state is open drain.

$\overline{CTSA}$, $\overline{CTSB}$. *Clear To Send* (inputs, active Low). When programmed as Auto Enables, a Low on these inputs enables the respective transmitter. If not programmed as Auto Enables, these inputs may be programmed as general-purpose inputs. Both inputs are Schmitt-trigger buffered to accommodate slow-risetime Signals. The Z80-SIO detects pulses on these inputs and interrupts the CPU on both logic level transitions. The Schmitt-trigger buffering does not guarantee a specified noise-level margin.

FIG. 11 — Z80-SIO/2 PIN CONFIGURATION



Z-80 SIO Bonding Options

DCDA, DCDB. *Data Carrier Detect* (inputs, active Low). These pins function as receiver enables i f the Z80-SIO is programmed for Auto Enables; otherwise they may be ured as general-purpore input pins. Both pins are Schmitt-trigger buffered to accommadate slow-risetime signals. The Z80-SIO detects pulse5 on there pins and interrupts Ihe CPU on both logic level transitions. Schmitt-trigger buffering d o e r not guarantee a specific noise level margin.

-- **RxDA, RxDB.** *Receive Data* (inputs, active High).

TxDA, TxDB. *Transmit Data* (outputs, active High).

RxCA, RxCB. *Receiver* Clocks (inputs). Receive data is sampled on Ihe rising edge of RxC. The Receive Clocks may be 1, 16, 32 or 64 rimes the dara rate in Asynchronous modes. These clocks may be driven by the Z80-CTC Counter Timer Circuit for programmable baud rate generation. Both inputs are Schmitt-trigger buffered (no noise level margin is specified). See the following section for bonding options.

TxCA, TxCB. *Transmitter* C l o c k s (inputs). TxD changes on Ihe falling edge of TxC. In Asynchronous modes, the Transmitter Clocks may be 1, 16, 32 or 64 times the data rate; however, the clock multiplier for Ihe transmitter and the receiver must be the same. The Trammit Clock inputs are Schmitt-trigger buffered for relaxed rise- and fall-time requiremenrs (no noise level margin is specified). Transmitter Clocks may be driven by the Z80-CTC Counter Timer Circuit for programmable baud rate generation. See the following section for bonding options.

RTSA, RTSB. *Request To Send* (outputs, active Low). When Ihe RTS bil is set, the RTS output goes Low. When the RTS bit is reset in Ihe Asynchronous mode. the output goes High after the transmitter is empty. In Synchronous modes. Ihe RTS pin strictly follows Ihe state of the RTS bit. Both pins can be used as general-purpose outputs.

DTRA, DTRB. *Data Terminal Ready* (outputs, active Low). See note on honding options These outputs follow the state programmed into the DTR bit. They can also he programmed as general-purpose outputs.

SYNC A, SYNC B. *Synchronization* (inputs/outputs, active Low). These pins can acr either as inputs or outputs. In the Asynchronous Receive mode, they are inputs similar Io CTS and DCD. In this mode. the transitions on these lines affect rhe state of the Sync/Hunt status bits in RR0. In Ihe External Sync mode, these lines also act as inputs. When external synchronization is achieved. SYNC must be driven Low on the second rising edge of RxC after that rising edge of RxC on which the last bit of the sync character was received. In other words, after the sync pattern is detected, the external logic must uait for two full Receive Clock cycles 10 activate Ihe SYNC input. Once SYNC is forced Low, it is wise 10 keep it Low until the CPU informs the external sync logic that synchronization has been lost or a new message is about to start. Character assembly begins on the rising edgc of RxC that immediately precedes the falling edge of SYNC in Ihe External Sync mode.

In Ihe Internal Synchronization mode (Monosync and Bisync), these pins act as outputs that are active during the part of Ihe receive clock (RxC) cycle in which sync characters are recognized. The sync condition is not latched, 50 Ihese outputs are active each time a sync pattern is recognized. regardless of character boundarier.

Bonding Options

The constraints of a 40-pin package make it impossible Io bring out the Receive Clock, Transmit Clock, Data Terminal Ready and Sync signals for both channels. Therefore. Cbannel B must sacrifice a signal or have two Signals bonded together. Since user rcquirements vary, three bonding options are offered:

- Z80-SIO/0 has all four Signals, but TxCB and RxCB are bonded together (Fig. 9).
- Z80-SIO/1 sacrifices DTRB and keeps TxCB, RxCB and SYNCB (Fig. IO).
- Z80-SIO/2 sacrifices SYNCB and keeps TxCB, RxCB and DTRB (Fig. 11).

Architecture

The device internal structure includes a Z80-CPU interface, internal control and interrupt logic, and two full-duplex channels. Each channel contains read and write registers, and discrete control and status logic that provides the interface to modems or other external devices.

The read and write register group includes five 8-bit control registers, two sync-character registers and Iwo status registers. The interrupt vector is written into an additional 8-bit register (Write Register 2) in Channcl B that may be read through Read Register 2 in Channel B. The registers for both channels are designated in the Iext as follows:

WR0-WR7 — Write Registers 0 through 7
RR0-RR2 — Read Registers 0 through 2

The bit assignment and functional grouping of each register is configured Io simplify and organize the programming process. Table 1 lists the functions assigned to each read or write register.

RR0	Transmit/Receive buffer status, interrupt status and external status
RR1	Special Receive Condition status
RR2	Modified interrupt vector (Channel B only)

Read Register Functions

Z-80 SIO Architecture

WR0	Register pointers, C R C initialize, initialization commands for the various modes, etc
WR1	Transmit/Receive interrupt and data transfer mode definition.
WR2	Interrupt vector (Channel B only)
WR3	Receive parameters and control
WR4	Transmit/Receive miscellaneous parameters and modes
WR5	Transmit parameters and controls
WR6	Sync character or SDLC address field
WR7	Sync character or SDLC flag

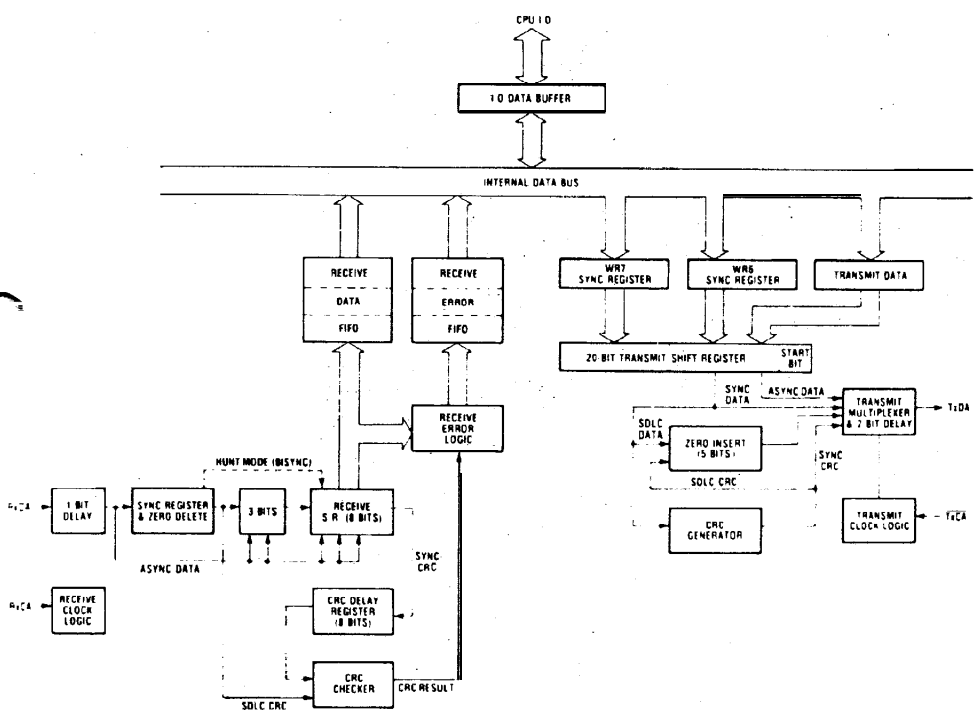
Write Register Functions

Table 1. Functional Assignments of Read and Write Registers

The logic for both channels provides formats, synchronization and Validation for data transferred to and from the channel interface. The modern control inputs Clear to Send ($\overline{CTS}$) and Data Carrier Detect ($\overline{DCD}$) are monitored by the discrete control logic under program control. All the modern control Signals are general purpose in nature and can be used for functions other than modern control.

For automatic interrupt vectoring, the interrupt control logic determines which channel and which device within the channel has the highest priority. Priority is fixed with Channel A assigned a higher priority than Channel B; Receive, Transmit and External Status interrupts are prioritized in that order within each channel.

FIG. 12 -TRANSMIT AND RECEIVE DATA PATH



Z 80-SIO
Z 80A-SIO

'Z-80 SIO Architecture

Data Path

The transmit and rceive dara path illustrated for Channel A in Figure1 2 is identical for both channels. The receiver has three 8-bit buffer registers in a FIFO arrangement in addition to the 8-bit receive shift register. This scheme creates additional time for the CPU to service an interrupt at the bepinning of a block of high-speed data. Incoming data is routed through one of several paths (data or CRC) depending on the selected mode and-in Asynchronous modes-the Character length.

The transmitter has an 8-bit transmit data register that is loaded from the internal data bus. and a 20-bit transmit shift register thar can be loaded from the sync Character buffers (WR6 and WR7) or from the transmit data register. Depending on the operational mode, outgoing data is routed rhrough one of four main paths before it is transmitted from the Transmit Data Output (TxD).

Functional Description

The funcrional capabilities of the Z80-SIO can be described from two different points of view: as a data communications device, it transmits and receives serial data, and meets the requirements of various data communications protocols; as a 280 family peripheral. it interacts with the Z80-CPU and other Z80 peripheral circuits, and shares the data, address and control busses, as well as being a part of the Z80 interrupt structure. As a peripheral to other microprocessors, the Z80-SIO offers valuable features such as non-vectored interrupts, polling and simple handshake capability.

The first part of the following functional description describes the interaction between the CPU and Z80-SIO; the second part introduces its data communications capabilities.

I/O Interface Capabilities

The Z80-SIO offers the choice of Polling. Interrupt (vectored or non-vectored) and Block Transfer modes to transfer data, status and control information to and from the CPU. The Block Transfer mode can be implemented under CPU or DMA control.

Polling. There are no interrupts in the Polled mode. Status registers RR0 and RR1 are updated at a,ppropriate times for each function being performed (for example, CRC Error status valid at the end of the message). All the interrupt modes of the Z80-SIO must be disabled to operate the device in a polled environment.

While in its Polling sequence, the CPU examines the status contained in RR0 for each channel; the RR0 status bits serve as an acknowledge to the Poll inquiry. The two RR0 status bits D_0 and D_2 indicate that a dara transfer is needed. The status also indicates Error or other special status conditions (see "Z80-SIO Programming"). The Special Receive Condition status contained in RR1 does not have to he read in a Polling requence because rhe status bits in RR1 must hc accompanied by a Receive Character Available status in RR0.

Interrupts. The Z80-SIO offers an elaborate interrupt scheme to provide fast interrupt response in real-time applications. Channel B regirters WR2 and RR2 contain the interrupt vector that points to an interrupt service routine in the memory. To service operations in both channels and to eliminate the necessity of writinp a status analysis routine, the Z80-SIO can modify the interrupt vector in RR2 so it points directly to one of eight interrupt service routines. this is done under program control by setting a program bit (WR1.D_2) in Channel B called "Status Affects Vector." When this bit is set, the interrupt vector in WR2 is modified according to the assigned priority of the various interrupting conditions. The table in the Write Register 1 description (Z80-SIO Programming section) shows the modiftcation details.

Transmit interrupts, Receive interrupts and External/Status interrupts are the main sources of interrupts. Each interrupt source is enabled undir program control with Channel A having a higher priority than Channel B, and with Receiver, Transmit and External/Status interrupts prioritized in that order within each channel. When the Transmit interrupt is enabled, the CPU is 'interrupted by the transmit buffer *becoming* empty. (This implies that the transmitter must have had a data character written into it so it can become empty.) When enabled, the receiver can interrupt the CPU in one of three ways:

- Interrupt on the first reteived Character
- Interrupt on all received characters
- Interrupt on a Special Receive condition

Interrupt On First Character is typically used with rhe Block Transfer mode. Interrupt On All Receive Characters has the option of modifying the interrupt vector in the event of a parity error The Special Receive Condition interrupt can occur on a character or message basis (End Of Frame interrupt in SDLC, for example). The Special Receive condition can cause an interrupt only if the Interrupt On First Receive Character or Interrupt On All Receive Characters mode is selected. In Interrupt On First Receive Character, an interrupt can occur from Special Receive conditions (except Parity Error) after the first receive character interrupt (example: Receive Overrun interrupt).

The main function of the External/Status interrupt is to monitor the signal transitions of the $\overline{CTS}$, $\overline{DCD}$ and SYNC pins; however, an External/Status interrupt is also caused by a Transmit Underrun condition or by the detection of a Break (Asynchronous mode) or Abort (SDLC mode) sequence in the data stream. The interrupt caused by the Break/Abort sequence has a special feature thar allows the Z80-SIO to interrupt when the Break/Abort sequence is detected or terminated. This feature facilitates the proper termination of the current message, correct initialization of the next message, and

Functional Description

thc **accurate** timing of the **Break/Abort condition** in **ex**ternal logic.

CPU/DMA Block Transfer. The Z80-SIO provides a Block Transfer **mode** to accommodate CPU block **trans**fer functions and DMA Controllers (Z80-DMA or other **designs)**. The Bloch Transfer mode uses the WAIT/READY output in conjunction with the Wait/Ready bits of Write Register 1. The WAIT/READY output can be dcfined **under**software control as a WAIT line in thc CPU. Block Transfer mode **or** as a READY line in the DMA Block Transfer mode.

To a DMA Controller, **the** Z80-SIO READY output indicates that the Z80-SIO is ready to transfer data to or from memory. To the CPU, the WAIT output indicates that thc Z80-SIO is not ready to transfer data, thereby **requesting** the CPU to extend the I/O cycle. The **programming** of bits 5, 6 and 7 of Write Register 1 and the logic states of the WAIT/READY line are defined in the Writc Register 1 description (Z80-SIO Programming **scction).**

Data Communicntions Cnpnbilities

In addition to thc I/O capabilities previously discussed, the **Z80-SIO** providcs **two** independent **full-duplex** channels that **can be programmed** for use in **Asynchronous, Synchronous** and SDLC (HDLC) modes. These different **modes** are **provided to** facilitate the **implementation** of commonly uscd data communications protocols. The following is a short description of the data **communications** protocols supported by the Z80-SIO. A **more** dctailcd explanation of these modes **can** be **found** in thc ***Z80-SIO Technical Manual.***

Asynchronous Modes. The Z80-SIO **offers** transmission and **reception** of five to eight bits per Character, plus optional evcn or odd parity. The transmitter **can** supply **one,** one and a half or two stop bits per Character and **can** provide a break output at any time. The **receiver** break **detection logic interrupts** the CPU only at the start **and** end of a received break. **Reception** is protected from **spikes** by a **transient spike** rejection mechanism that **checks** thc **signal one-half** a bit **time** after a Low **level** is **detected on** the Receive Data input. **If** the Low **does** not persist-as in the **case** of a transicnt-the **character** asscmbly process is not started.

Framing errors and **overrun** crrors arc detected and **buffered** rogcther with **thc** partial Character **on which** thcy occurred. Vectored interrupts **allow** fast **servicing** of error conditions using dedicated routincs. **Furthermore,** a **built-in** chccking process avoids interpreting a framing **error as** a **new start** bit: a framing **error** results in the addition of one-half a bit time **to** thc **point** at **which** thc search for the next starr bit is **begun.**

The **Z80-SIO does** not rcquire symmctric Transmit and Rcccivc **Clock** Signals-a feature that **allows** it **to be used** with a Z80-CTC **or** any other **clock** Source. The **transmitter** and rccciver **can** handle data at a rate of **1, 1/16, 1/32** or **1/64** of the **clock** rate **supplied to** thc Rcccivc and Transmit **Clock** inputs.

In Asynchronous **modes,** the SYNC pin may be **pro**grammed for an input **that can** bc used for functions such **as** monitoring **a** ring **indicator.**

Synchronous Modes. The Z80-SIO supports both byte-oricnted and bit-oriented synchronous communication. Synchronous byte-oriented **protocols can** be handled in several **modes** that **allow Character** synchronization with an S-bit sync Character **(Monosync),** any 16-bit sync pattern (Bisync), or with an **external** sync signal **Leading** sync **characters can** he removed uithout intcrrupting the CPU. CRC checking for synchronous byte-oriented modes is delayed by one Character time so the CPU may ditable CRC checking on tpecific characters. **This** permit**s implementation** of protocols such as IBM Bisync.

Both CRC-16 ($X^{16}+X^{15}+X^2+1$) and CCITT ($X^{16}+X^{12}+X^5+1$) error **checking** polynomials are supported. In all **non-SDLC** modes, the CRC generator is **initialized to 0's;** in SDLC modes, it is initializcd to 1's. **(This** means that the **Z80-SIO cannot** generate or check CRC for IBM-**compatible** soft-sectorcd disks.) The Z80-SIO also provides a feature that automarically transmits CRC data **when** no other data is available for transmission. This **allows** very high-speed transmissions undcr DMA control with no need for CPU intervention at the end of a message. When therc is **no data or** CRC to send in **Synchronous** modes, the transmitter inserts **8-or** 16-bit sync **characters** regardlcss of the programmed Character length. Sincc the **CPU can** read **status information** from the **Z80-SIO,** it **can** determine thc **type** of transmission **(data,** CRC or sync **characters)** that is **taking place** at any time.

The Z80-SIO supports **synchronous** bit-oriented protocols such as SDLC and HDLC by performing **automatic** flag scnding, zero insertion and CRC generation. A **special command can** be uscd to **abort** a frame in **transmission.** The **Z80-SIO** automatically transmits the CRC and trailing **flag** when the transmit buffer **becomes** empty. An **interrupt** warns thc CPU of **this status change** so an **abort** may be issucd if a transmitter undcrrun has **occurred.** Onc to eight bits per Character **can** be sent, **which allows** transmission of a message exactly as received with no **prior** information **about** the Character **structure** in the information field of a **frame.**

The **receiver** automatically **synchronizes on** the **leading** flag of a frame and provides a **synchronization** signal **that can** bc **programmed to interrupt.** In addition, an **interrupt** on the first rcceivcd Character **or** on **every** Character **can** be selectcd. The rcccivcr automatically **deletes** all **zeroes** inserted by the **transmitter during character** assembly. It also calculars and automatically **checks** the CRC to validate **frame** transmission. At **the** end of transmission. the **status** of a received **frame** is availablc in thc **status registers.** The **receiver can** be **programmed to** scarch for frames addressed to only a **specified** user-sclectablc address or to **a** global broadcast **address.** In this **mode, frames** that do not **match** thc **user-**

selected or broadcast addrcss arc ignored. The Address Search mode providcs for a singlc-bytc address recognizable by thc hardware. The number of addrcss bytcs can be extcndcd undergoftware control.

The Z80-SIO can bc convcnicntly uscd under DMA control to providc high-speed rcccption. The Z80-SIO can interrupt the CPU whcn the first Character of a message is rcccived. The CPU then cnables thc DMA to transfer the messagc to memory. The Z80-SIO then issues an End Of Frame interrupt and the CPU checks the status of the receivcd mcssage. Thus, thc CPU is freed for other service while thc mcssage is being ieceived. A similar schcme allows message tiansmission undcr DMA control.

Z80-SIO Programming

To program the Z80-SIO, thc system program first issues a series of commands that initialize the basic mode of Operation and then other commands that qualify conditions within thc selected mode. For cxamplc, the Asynchronous mode. Character length, clock rate, number of stop bits, even or odd pririty are first set, then thc interrupt mode and, finally, receiver or transmitter cnablc. The WR4 Parameters must be issued bcforc any othcr Parameters arc issued in thc initialization routine.

Borh channcls contain command registers that must bc programmed via the system program prior to operation. The Channel Select input ($B/\overline{A}$) and the Control/Data input ($C/\overline{D}$) are the command structure addressing controls, and are normally controlled by the CPU address bus. Figure 14 illustratcs thc timing rclationships for programming the write registers, and transferring data and Status.

Write Registers .

The Z80-SIO contains cight registers (WR0-WR7) in each channcl that are programmed separately by the system program to configure thc functional personality of thc channels. With thc exception of WR0, programming thc write registers requires two bytcs. The first byte contains three bits (D_0-D_2) that point to the selectcd register; the sccond byre is thc actual control word that is writtcn into the register to configure thc Z80-SIO.

WR0 is a special case in that all thc basic commands (CMD_0-CMD_2) can be acccsscd with a single byte. Reset (internal or external) initializes thc poinlcr bits D_0-D_2 10 point to WR0.

Resd Registers

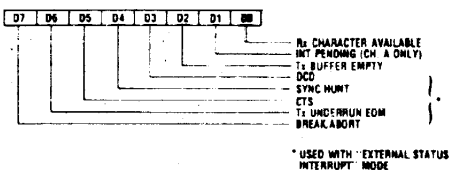
The Z80-SIO contains three registers, RR0-RR2 (Figure 12). that can bc read to obtain thc status information for each channcl (except for RR2 — Channcl B only). The status information includcs error conditions, interrupt vector and standard communications-interface Signals.

To read the contents of a selected read register other than RR0, the system program must first write the pointer byte to WR0 in exactly thc ramc way as a writc register operation. Then, by executing an input instruction, thc contents of the addressed read register can be read by the CPU.

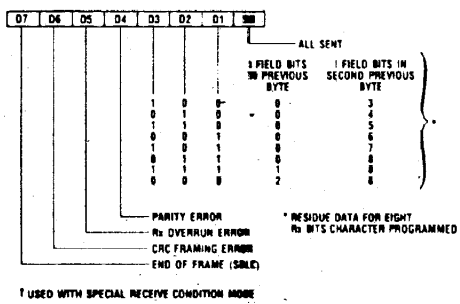
The status bits of RR0 and RR1 are carcfully groupcd to simplify status monitoring. For example. when the interrupt vector indicates that a Special Rcceive Condition iatcrrupt has occurrcd, all the appropriate error bits can be read from a Single register (RR1).

FIG. 12 — READ REGISTER BIT FUNTIONS

READ REGISTER 0



READ REGISTER 1 †


D7 D6 D5 D4 D3 D2 D1 D0
ALL SENT
3 FIELD BITS IN PREVIOUS BYTE
3 FIELD BITS IN SECOND PREVIOUS BYTE
PARITY ERROR
Rx OVERRUN ERROR
CRC FRAMING ERROR
END OF FRAME (SDLC)
* RESIDUE DATA FOR EIGHT Rx BITS CHARACTER PROGRAMMED
† USED WITH SPECIAL RECEIVE CONDITION MODE

READ REGISTER 2

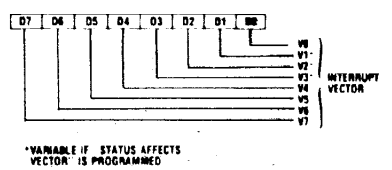
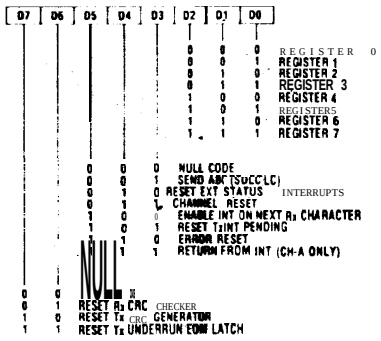


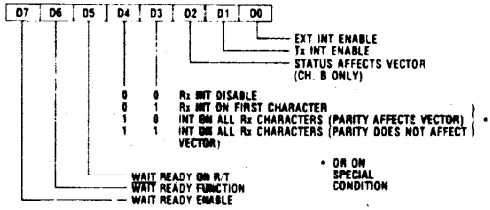
FIG. 13 — WRITE REGISTER BIT FUNCTIONS

Z-SO SIO Programming

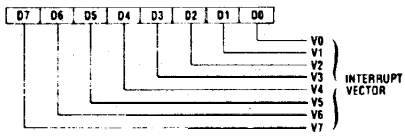
WRITE REGISTER 0



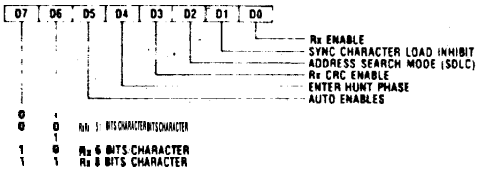
WRITE REGISTER 1


D7 D6 D5 D4 D3 D2 D1 D0
EXT INT ENABLE
Tx INT ENABLE
STATUS AFFECTS VECTOR (CH. B ONLY)
Rx INT DISABLE
Rx INT ON FIRST CHARACTER
INT ON ALL Rx CHARACTERS (PARITY AFFECTS VECTOR)
INT ON ALL Rx CHARACTERS (PARITY DOES NOT AFFECT VECTOR)
* OR ON SPECIAL CONDITION
WAIT READY ON R/T
WAIT READY FUNCTION
WAIT READY ENABLE

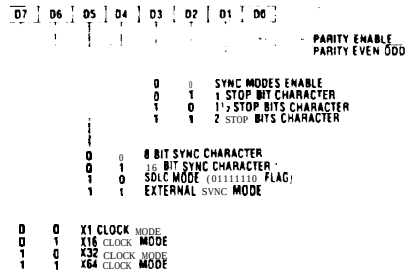
WRITE REGISTER 2 (CHANNEL B ONLY)



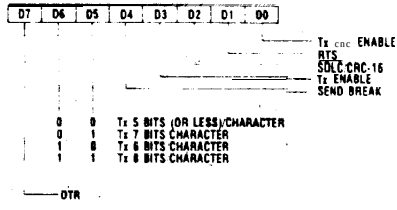
WRITE REGISTER 3



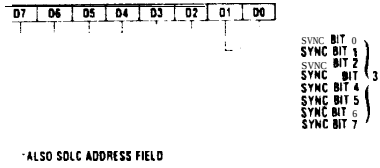
WRITE REGISTER 4



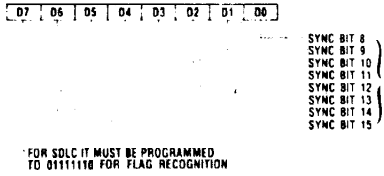
WRITE REGISTER 5



WRITE REGISTER 6



WRITE REGISTER 7


D7 D6 D5 D4 D3 D2 D1 D0
SYNC BIT 8
SYNC BIT 9
SYNC BIT 10
SYNC BIT 11
SYNC BIT 12
SYNC BIT 13
SYNC BIT 14
SYNC BIT 15
*FOR SDLC IT MUST BE PROGRAMMED TO 01111110 FOR FLAG RECOGNITION

Z-80 SIO Timing

Timing

Rend Cycle. The timing signals gcnerated by a ZOO-CPU input instruction Io read a Data or Status byte from the Z80-SIO are illustrated in Figure 14a.

Write Cycle. Figure 14b illustrates the timing and dara signals generated by a Z80-CPU output instruction to write a Data or Control byte into the Z80-SIO.

FIG. 14a – READ CYCLE

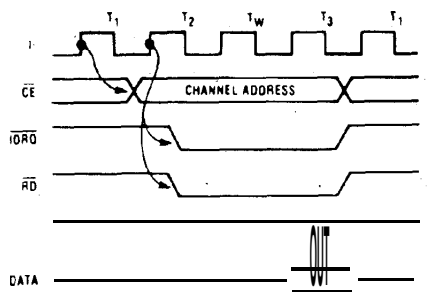
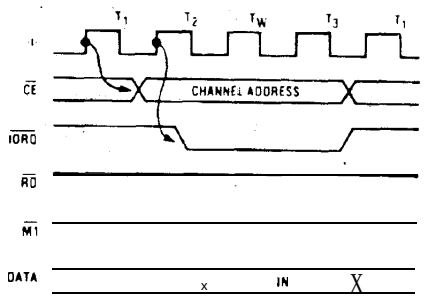


FIG. 14b – WRITE CYCLE



Interrupt Acknowledge Cycle. After receiving an Interrupt Request signal (INT pulled Low), the Z80-CPU sends an Interrupt Acknowledge signal (M1 and IORQ both Low). The daisy-chained interrupr circuits determine the highest priority interrupt requestor. The IEI of the highest priority peripheral is terminated High. For any peripheral that has no interrupt pending or under Service. IEO = IEI. Any peripheral that does have an interrupt pending or under service forces its IEO Low.

To insure stable conditions in the daisy chain, all interrupt status Signals are prevented from changing while M1 is Lou. When IORQ is Low, the highest priority interrupt requestor (Ihe one with IEI High) places its interrupt vector on the data bus and sets its infernal interrupt-under-service latch.

Return From Interrupt Cycle. Normally, the Z80-CPU issues a RETI (RETurn from Interrupt) instruction at the end of an interrupt service routine. RETI is a 2-byte opcode (ED-4D) that resets the interrupt-under-service latch to terminate the interrupt that has just been processed. This is accomplished by manipulating the daisy chain in Ihe folloring way.

The normal daisy chain Operation can be used to detect a pending interrupt; however, it cannot distinguish between an interrupt under service and a pending unacknowledged interrupt of a higher priority. Whenever "ED" is decoded, Ihe daisy chain is modified by forcing High the IEO of any interrupt that has not yet been acknowledged. Thus the daisy chain identifies the device presently under Serviceas the only one with an IEI

FIG. 14c – INTERRUPT ACKNOWLEDGE CYCLE

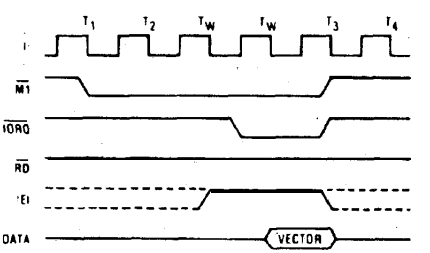
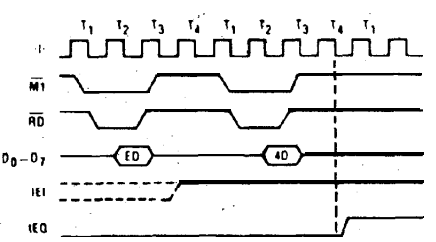


FIG. 14d – RETURN FROM INTERRUPT CYCLE



Z-80 SIO Timing

High and an IEO Low. If the next opcode byte is "4D," the interrupt-under-service latch is reset.

The ripple time of thc irterrupt daisy chain (both rhe High-to-Low and the Low-to-High transitions) limits the number of devices that can be placed in the daisy chain. Ripple time can be improved with carry-look-ahead, or by extending the interrupt acknowledge cycle.

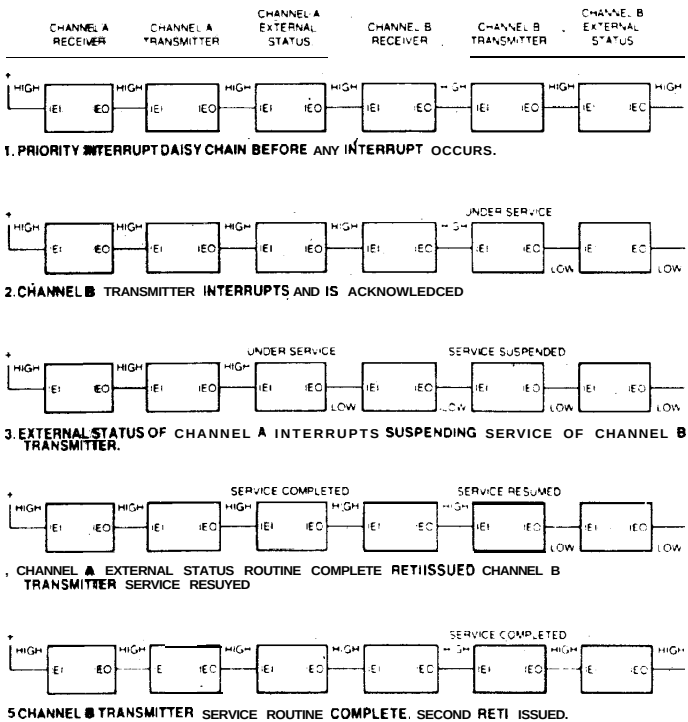
Daisy Chain Interrupt Nesting

Figure 15 illustrates the daisy chain configuration of interrupt circuits and their behavior with nested interrupts (an interrupt that is interrupted by another with a higher priority).

Each box in the illustration could be a separate external Z80 peripheral circuit with a user-defined order of interrupt priorities. However. a similar daisy chain structure also exists inside the Z80-SIO, which has six interrupt levels with a fixed order of prioriries.

The case illustrated occurs when the transmitter of Channel B interruprs and is granted Service. While this interrupt is being serviced, it is interrupted by a higher priority interrupt from Channel A. The second interrupt is servicccd and-upon completion-a RETI instruction is executed or a RETI command is written into the Z80-SIO, resetting the interrupt-under-service latch of the Channel A interrupt. At this time, the service rotine for Channel B is resumed. When it is completed, another RETI instruction is executed to complete the interrupt Service.

FIG. 15 – TYPICAL INTERRUPT SEQUENCE



1. PRIORITY INTERRUPT DAISY CHAIN BEFORE ANY INTERRUPT OCCURS.

2. CHANNEL B TRANSMITTER INTERRUPTS AND IS ACKNOWLEDCED

3. EXTERNAL STATUS OF CHANNEL A INTERRUPTS SUSPENDING SERVICE OF CHANNEL B TRANSMITTER.

, CHANNEL A EXTERNAL STATUS ROUTINE COMPLETE RETIISSUED CHANNEL B TRANSMITTER SERVICE RESUYED

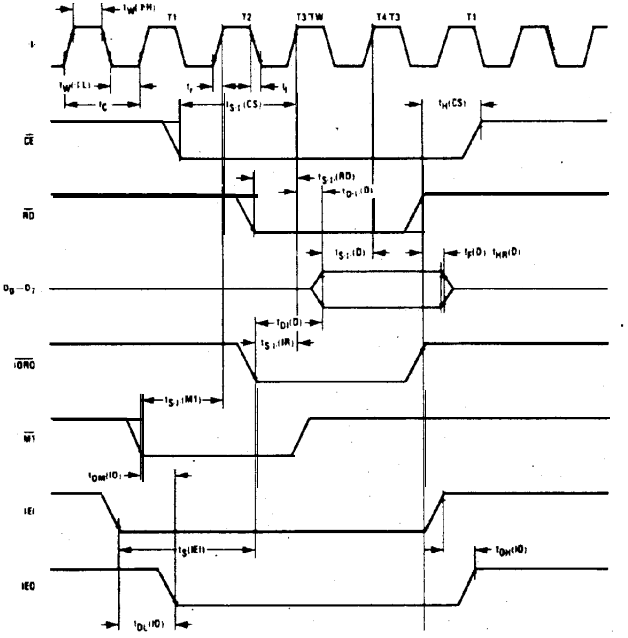
5 CHANNEL B TRANSMITTER SERVICE ROUTINE COMPLETE. SECOND RETI ISSUED.

Electrical Characteristics

FIG. 16 — TYPICAL INTERRUPT SEQUENCE

AC Characteristics

$T_A = 0°C$, $V_{CC} = +5V, \pm 5\%$

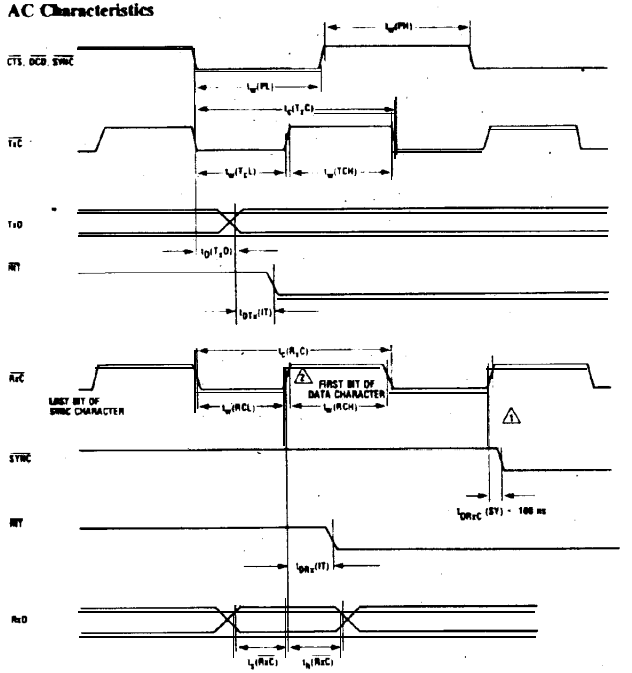


Signal	Symbol	Parameter	Z80-SIO Min	Z80-SIO Max	Z80A-SIO Min	Z80A-SIO Max	Unit
	$t_{c(\phi)}$	Clock Period	400	4000	250	4000	ns
	$t_{w(\phi H)}$	Clock Pulse Width: clock HIGH	170	2000	105	2000	ns
ϕ	$t_{w(\phi L)}$	Clock Pulse Width: clock LOW	170	2000	105	2000	ns
	t_r, t_f	Clock Rise and Fall Times	0	30	0	30	ns
	t_s	Any Unspecified Hold Time for setup times specified below	0		0		ns
$\overline{CE}$, $\overline{B/A}$, $\overline{C/D}$, $\overline{IORQ}$	$t_{s(CS)}$	Control Signal Setup Time to rising edge of ϕ during Read or Write Cycle	160		145		ns
D_0–D_7	$t_{d(D)}$	Data Output Delay from rising edge of ϕ during Read Cycle		240		220	ns
	$t_{s(D)}$	Data Setup Time to rising edge of ϕ during Write or M1 Cycle	50		50		ns
	$t_{d(D)}$	Data Output Delay from falling edge of $\overline{IORQ}$ during INTA Cycle		340		160	ns
	$t_{z(D)}$	Delay to Floating Bus (output buffer disable time)		230		110	ns
IEI	$t_{s(IEI)}$	IEI Setup Time to falling edge of $\overline{IORQ}$ during INTA Cycle	200		140		ns
IEO	$t_{dr(IO)}$	IEO Delay Time from rising edge of IEI (after ED decode)		150		100	ns
	$t_{df(IO)}$	IEO Delay Time from falling edge of IEI		150		100	ns
	$t_{dM(IO)}$	IEO Delay Time from falling edge of $\overline{M1}$ (interrupt occurring just prior to M1)		300		190	ns
$\overline{M1}$	$t_{s(M1)}$	$\overline{M1}$ Setup Time to rising edge of ϕ during INTA or M1 Cycle	210		90		ns
$\overline{RD}$	$t_{s(RD)}$	$\overline{RD}$ Setup Time to rising edge of ϕ during Read or M1 Cycle	240		115		ns

*If WAIT from the SIO is to be used, $\overline{CE}$, $\overline{IORQ}$, C $\overline{D}$ and $\overline{M1}$ must be valid for as long as the Wait condition is to persist

Electrical Characteristics

AC Characteristics



NOTES:

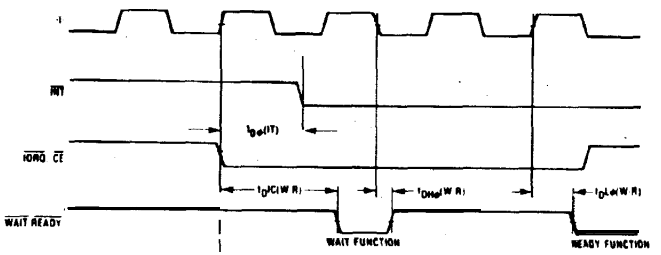
1. The SYNC input must be driven Low on the rising edge of RxC delayed two complete clock cycles from the last bit of the sync character.
2. Data character assembly begins on the next Receive Clock cycle after the last bit of the sync character is received.

Signal	Symbol	Parameter	Z80-SIO Min	Z80-SIO Max	Z80A-SIO Min	Z80A-SIO Max	Unit
INT	$t_{dis}(IT)$	INT Delay Time from rising edge of RxC	10	13	10	13	ϕ periods
	$t_{dr}(IT)$	INT Delay Time from transition of Xmit Data Bit	5	9	5	9	ϕ periods
CTSA CTSB DCDA DCDB SYNCA SYNCB	$t_w(PH)$	Minimum HIGH Pulse Width for latching state into register and generating interrupt	200		200		ns
	$t_w(PL)$	Minimum LOW Pulse Width for latching state into register and generating interrupt	200		200		ns
SYNCA SYNCB	$t_{ds}(SY)$	Sync Pulse Delay Time from rising edge of RxC Output Modes	4	7	4	7	ϕ periods
	$t_{dss}(SY)$	Sync Pulse Delay Time from rising edge of RxC External Sync Mode		100		100	ns
TxCA TxCB	$t_c(TxC)$	Transmit Clock Period	400	*	400	*	ns
	$t_w(TCH)$	Transmit Clock Pulse Width, clock HIGH	180	*	180	*	ns
	$t_w(TCL)$	Transmit Clock Pulse Width, clock LOW	180	*	180	*	ns
TxDA TxDB	$t_d(TxD)$	TxD Output Delay from falling Edge of TxC (x1 Clock Mode)		400		300	ns
RxCA RxCB	$t_c(RxC)$	Receive Clock Period	400	*	400	*	ns
	$t_w(RCH)$	Receive Clock Pulse Width, clock HIGH	180	*	180	*	ns
	$t_w(RCL)$	Receive Clock Pulse Width, clock LOW	180	*	180	*	ns
RxDA RxDB	$t_s(RxC)$	Setup Time to rising edge of RxC, x1 mode	0		0		ns
	$t_h(RxC)$	Hold Time from rising edge of RxC, x1 mode	140		140		ns

*In all modes, the system clock (ϕ) rate must be at least 4.5 times the maximum data rate. RESET must be active a minimum of one complete ϕ cycle.

Electrical Characteristics

AC Charactcristics



Signal	Symbol	Parameter	Z80-SIO Min	Z80-SIO Max	Z80A-SIO Min	Z80A-SIO Max	Unit
INT	$t_{\phi I(IT)}$	INT Delay Time from rising edge of ϕ		200		200	ns
WAIT READY	$t_{\phi C(W/R)}$	WAIT READY Delay Time from IORQ or CE in Wait Mode		300		210	ns
	$t_{\phi H(W/R)}$	WAIT READY Delay Time from falling edge of ϕ, WAIT READY HIGH Wait Mode		150		130	ns
	$t_{\phi R(W/R)}$	WAIT READY Delay Time from rising edge of RxC Data Bit Ready Mode	10	13	10	13	ϕ periods
	$t_{TR(W/R)}$	WAIT READY Delay Time from center of Transmit Data Bit Ready Mode	5	9	5	9	ϕ periods
	$t_{\phi L(W/R)}$	WAIT READY Delay Time from rising edge of ϕ, WAIT READY LOW Ready Mode		120		120	ns

DC Characteristics

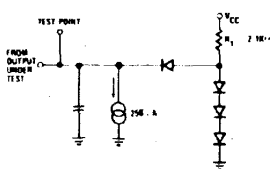
$T_A = 0°C$ to $70°C$, $V_{CC} = +5V, \pm 5\%$

Symbol	Parameter	Min.	Max	Unit	Test Condition
V_{ILC}	Clock Input Low Voltage	−0.3	+0.45	V	
V_{IHC}	Clock Input High Voltage	$V_{CC}-0.6$	+5.5	V	
V_{IL}	Input Low Voltage	−0.3	+0.8	V	
V_{IH}	Input High Voltage	+2.0	+5.5	V	
V_{OL}	Output Low Voltage		+0.4	V	$I_{OL} = 2.0$ mA
V_{OH}	Output High Voltage	+2.4		V	$I_{OH} = -250$ μA
I_{LI}	Input Leakage Current	−10	+10	μA	$0 \le V_{in} \le V_{CC}$
I_Z	3-State Output/Data Bus Input Leakage Current	−10	+10	μA	$0 \le V_{in} \le V_{CC}$
I_{LSY}	SYNC Pin Leakage Current	−40	+10	μA	$0 \le V_{in} \le V_{CC}$
I_{CC}	Power Supply Current		100	mA	

Capocitance

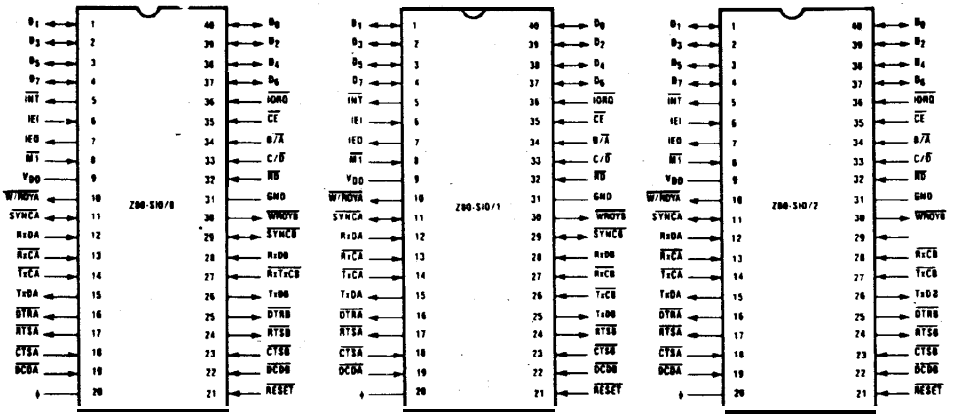
$T_A = 25°C$, f= 1 MHz

Symbol	Parameter	Min.	Max.	Unit	Test Condition
C	Clock Capacitance		40	pF	Unmeasured pins returned to ground
C_{IN}	Input Capacitance		5	pF	
C_{OUT}	Output Capacitance		10	pF	



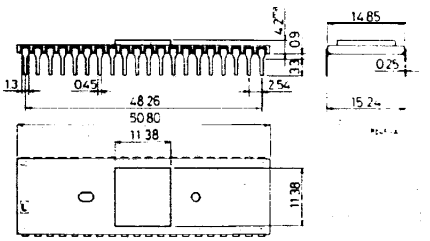
Z 80-SIO
Z 80A-SIO

Package Information

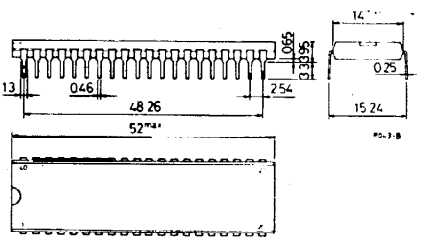


MECHANICAL DATA (dimensions in mm)

40-PIN CERAMIC DUAL IN-LINE SLAM PACKAGE



40-PIN PLASTIC DUAL IN-LINE PACKAGE



ORDERING NUMBERS:

Z80-SI0 **D1** for dual in-he ceramic slam package
Z8O-SIO Bl for dual in-line plastic package
Z8OA-SIO D1 for dual in-line ceramic slam package
Z8OA-SI0BI for dual in-line plastic package